Abhrajyoti Kundu
Computer Science & IT (CS)

OPERATING SYSTEM (GATE 2023) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

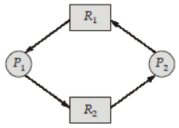
ALL(33) | CORRECT(23) | INCORRECT(9) | SKIPPED(1)

Q. 1 Have any Doubt ?

While of the following is incorrect?

A. If a system uses preemptive CPU schedular then deadlock is not possible in system. — Your answer is **Correct**

Solution :
(a)
Preemptive CPU schedular has nothing to do with deadlock. There can be dead state for this system.



B. Deadlock state is an unsafe state.

C. If a system has total ordering of all resource types and each process requests resources only in a increasing order of enumeration, then system will never go in deadlock state.

D. If a system is in deadlock state and all deadlocked processes aborted, then system will recover from deadlock.

QUESTION ANALYTICS +

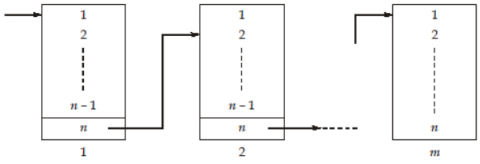
Q. 2 Have any Doubt ?

Consider a file system which uses grouping method for free-space management.
Disk block size is 4 KB and disk block address is 32 bit long. Assume, at particular instance disk has 3 blocks which used for grouping. The maximum free blocks available for a file at this instance are _____.

3070 — Correct Option

Solution :
3070
In grouping, if m blocks used and n in number of addresses per block, then in first $(m-1)$ blocks $(n-1)$ addresses point to free blocks and n^{th} address points to next blocks used for grouping.
And last (m^{th}) block can used completely.

$$\text{Block size (BS)} = 4\text{ KB}$$
$$\text{Block address} = 4\text{ B}$$
$$\text{Addresses per blocks } (n) = \frac{4\text{ KB}}{4} = 2^{10}$$
$$\text{Number of grouping blocks } (m) = 3$$
$$\text{Number of free blocks} = (n-1)(m-1) + n$$
$$= (2^{10}-1) \times 2 + 2^{10} = 3070$$



Your Answer is 2048

QUESTION ANALYTICS +

Q. 3 Have any Doubt ?

Consider a process *P* invokes the exec( ) system call on UNIX system. Which of following informations of Process Control Block (PCB) of P will change after successful invoke

of exec( )?

A Process identifier (PID)

B Value of program counter stored within user space context on Kernel stack — Your option is **Correct**

C PID of parent process

D Page table entries — Your option is **Correct**

YOUR ANSWER - b,d | CORRECT ANSWER - b,d | STATUS - ✔

Solution :

(b, d)
exec( ) system call used to replace the process's memory space with new program. New memory space will lead to new program counter, now page tables, etc.
PID of *P* and its parent have nothing to do with new program.

QUESTION ANALYTICS +

Q. 4

Have any Doubt ?

Consider a process *P* running on a system with non-preemptive Kernel design. Consider following actions by *P*:
(i) A blocking system call.
(ii) Servicing a timer interrupt.
(iii) Exit system call.
(iv) Servicing a disk interrupt, that results in another blocked process being marked as ready.
Which of the above actions always result in context-switch of *P*?

A (iii) only

B (i) and (iii) only — Your answer is **Correct**

Solution :
(b)
Exit system call definitely lead to context-switch. Same for blocking system call.
Timer interrupt cannot lead context-switch always.

C (i) and (ii) only

D (i), (ii), (iii) and (iv) all actions

QUESTION ANALYTICS +

Q. 5

Have any Doubt ?

Consider a process has following segment table:

Segment number	Base (decimal)	Limit/length (decimal)
0	300	400
1	1500	100
2	800	50
3	900	450
4	100	150

Logical address for above system represented as <segment number, offset>. Which of the following logical address for above process is invalid and lead to an error?

A <0, 301>

B <1, 89>

C <2, 845> — Your answer is **Correct**

Solution :
(c)
In (c), offset is 845 and limit for segment 2 is 50.
Here, offset > limit, it is invalid.

D <4, 149>

QUESTION ANALYTICS +

Q. 6

Have any Doubt ?

Consider a process with 5 frames and it has a page-reference string of length 100 (100 pages request in a sequence). The page-reference string has 10 distinct page numbers. Let n and m are the lower bound and upper bound on number of page faults respectively when process execute
given page-reference string. Operating system uses First-In-First-Out (FIFO) page-replacement algorithm. The possible value of $n + m$ is _____.
(Assume frames are empty initially)

110 — Your answer is Correct110

Solution :
110
String has 10 different pages numbers. And when a page number referenced first time it will be page fault.
So, lower bound on page faults (n) = 10.
String length is 100. There can be the situation where each reference is page fault. So, upper bound on page faults (m) = 100

$$n + m = 10 + 100 = 110$$

QUESTION ANALYTICS +

Q. 7

Have any Doubt ?

Consider following pseudo-codes for a process and TestAndSet function:

```
function TestAndSet (boolean *lock)
{
    boolean old = *lock
    *lock = FALSE;
    return old;
}
function process (int N)
{
    while (TRUE) {
        while (TestAndSet (lock));
        critical section;
        lock = TRUE;
        remainder section;
    }
}
```

Assume N processes, where N ≥ 2, concurrently executing with function process shown above.
Which of the following is incorrect for critical section solution in above situation?

A Processes satisfy mutual exclusion requirement. — Your answer is Correct

Solution :
(a)
If lock == FALSE, then all processes can go in critical section.
If lock == TRUE, then in next iteration of inner while lock = FALSE, and again all process can in critical section. So, mutual exclusion not satisfy.
Any process can enter any time in critical section. So, solution satisfy progressive and bounded waiting requirement.

B Critical section solution satisfy progressive requirement.

C Critical section solution satisfy bounded waiting requirement.

D None of above is incorrect.

QUESTION ANALYTICS +

Q. 8

Have any Doubt ?

Consider a system uses demand-paged memory. The page table is held in registers and registers take negligible time to access as compare to memory. Memory access time is 20 micro-seconds. Operating system takes 600 micro-seconds to service a page fault if an empty page is available or
the replace page is not modified and 1 milliseconds if the replaced page is modified. Assume replaced page is modified 60% of the time. The maximum acceptable page-fault rate for an effective access time of no more than 30 microseconds is _____. (Upto 3 decimal places)

0.012 — Your answer is Correct0.012

Solution :
0.012
Page table (PT) access time is 0, due to registers.
Let r is page fault rate.

$$\text{Effective access time} = (1 - r) \times (\text{PT access time} + \text{Memory access time}) + r \times (\text{Page fault service})$$
$$30\ \mu s = (1 - r)\ (0 + 20\ \mu s) + r(0.4 \times 600\ \mu s + 0.6 \times 1000\ \mu s)$$
$$= (1 - r)\ (20) + r(240 + 600)$$
$$30 - 20 = -20 \times r + r(840)$$
$$10 = r \times 820$$
$$r = \frac{1}{82} \approx 0.012$$

QUESTION ANALYTICS +

Q. 9 Have any Doubt ?

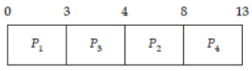
Consider the processes with arrival time and CPU burst time given in following table:

Process	Arrival Time (ms)	Burst Time (ms)
P_1	0	3
P_2	1	4
P_3	2	1
P_4	3	5

Schedular uses non-preemptive Shortest Job First (SJF) scheduling algorithm to schedule above processes. Average waiting time of above processes is _____ ms.

2.25 Your answer is **Correct**2.25

Solution :
2.25
Gantt chart for above processes:



Waiting time = Turn around time – Burst time
WT (P_1) = 3 – 3 = 0
WT (P_2) = 7 – 4 = 3
WT (P_3) = 2 – 1 = 1
WT (P_4) = 10 – 5 = 5

Average waiting time = $\frac{0+3+1+5}{4} = \frac{9}{4} = 2.25$

QUESTION ANALYTICS +

Q. 10 Have any Doubt ?

Consider a page table entry in page table of a process which has valid bit set and present bit unset. Which of the following is not correct event/output on memory access to virtual address belonging to this page table entry? (Assume system uses TLB)

A MMU generate trap to the OS

B TLB miss

C Page fault will occur

D TLB miss but no page fault will occur Your answer is **Correct**

Solution :
(d)
If valid bit is set then it is allocated virtual space, and if present bit unset then frame is not allocated to this address.
So, no entry in TLB (TLB miss). Then MMU go to page table entry (PTE), and present bit = 0, so
generate trap to OS.
Now, it is page fault.

QUESTION ANALYTICS +